

What is claimed is:

- 1) A method for creating a plurality of queues using a shared data buffer, the method comprising:
 - providing a plurality of pointers to the data buffer, each pointer associated with an area of the buffer; and
 - creating a given queue by associating a given pointer from the plurality of pointers with the given queue.
- 2) A method according to claim 1, wherein providing a plurality of pointers includes storing the plurality of pointers in a free pointer linked list.
- 3) A method according to claim 2, wherein associating the given pointer includes removing the given pointer from the free pointer linked list.
- 4) A method according to claim 3, wherein associating the given pointer further includes storing the pointer in a given queue linked list.
- 5) A method according to claim 4 further including:
 - removing the given pointer from the queue linked list and adding the given pointer to the free pointer linked list to delete a member of the given queue.
- 6) A method according to claim 5, wherein the given queue is a FIFO queue.
- 7) A method according to claim 5, wherein the given queue is a LIFO queue.
- 8) A method according to claim 4 wherein the free pointer linked list and the given queue linked list are stored in a given data structure.
- 9) A computer program product for use on a computer system for managing queues, employing a shared data buffer, the computer program product comprising a computer usable medium having computer readable program code thereon, the computer readable program code including program code for:
 - providing a plurality of pointers to the data buffer, each pointer associated with an area of the buffer; and
 - creating a given queue by associating a given pointer from the plurality of pointers with the given queue.
- 10) A computer program product according to claim 9, wherein providing a plurality of pointers includes storing the plurality of pointers in a free pointer linked list.

- 11) A computer program product according to claim 10, wherein associating the given pointer includes removing the given pointer from the free pointer linked list.
- 12) A computer program product according to claim 11, wherein associating the given pointer further includes storing the pointer in a given queue linked list.
- 13) A computer program product according to claim 12 further including:
removing the given pointer from the queue linked list and adding the given pointer to the free pointer linked list to delete a member of the given queue.
- 14) A computer program product according to claim 13, wherein the given queue is a FIFO queue.
- 15) A computer program product according to claim 13, wherein the given queue is a LIFO queue.
- 16) A computer program product according to claim 12 wherein the free pointer linked list and the given queue link list are stored in a given data structure.
- 17) A device for managing queues in a computer system, the device comprising:
a shared data buffer;
a pointer array pointing to a plurality of areas of the data buffer;
a free list data structure including an entry count, a head pointer to the data buffer and a tail pointer to the data buffer;
a queue state including a plurality of virtual queue data structures, each queue data structure including a queue entry count, a queue head pointer and a queue tail pointer, the queue head pointer and the queue tail pointer pointing to areas of the data buffer;
and
logic for deleting an entry from the free list data structure and adding the entry to a given virtual queue data structure.
- 18) A device according to claim 17, the device further comprising:
logic for deleting an entry from a given virtual queue data structure and adding the entry to the free list data structure.